Lists and More Iteration

CS101
Fall 2017
What are Lists?

A List is a mutable, ordered sequence of elements.

They are one of many Data Structures you will learn about – ways of storing and organizing data.
How do we make one?

Method 1:
Using the built-in list constructor

L = list()

OR

L = list([element 1, element 2, ...])
How do we make one?

Method 2:
Using the shorthand display notation (hard brackets)

L = []

OR

L = [element1, element2, ...]
How do we make one?

Method 3:
Using a list comprehension

\[ L = \text{list(expression for variable in sequence)} \]

OR

\[ L = [\text{expression for variable in sequence}] \]

(Yes, that is a for loop inside a list!)
What can we put in them?

ANYTHING, as long as it is a valid Python expression!

The list:

\[ L = [5, 10, "Hello", True and False, 5+3, "C"*3] \]

is valid syntax and evaluates to

\[ [5, 10, "Hello", False, 8, CCC] \]
What can we put in them?

You can even put a list inside a list (called “nesting”)!  

$L = [5, 6, [“the”, “inner”, “list”], 7, 8]$
How do we access them?

Indexing!

If we have list \( L = ["Cat", "dog", 10, 20, True] \)

- \( \text{len}(L) \) is 5
- Indices 0 to 4
- \( L[0] \) is "Cat"
- \( L[4] \) is True
- \( L[1:4] \) is ["dog", 10, 20]
How do we access them?

If we use the nested list from earlier:

\[ L = [5, 6, ["the", "inner", "list"], 7, 8] \]

\[ \text{len}(L) \text{ is } 5 \]
\[ \text{Indices } 0 \text{ to } 4 \]
\[ L[2] \text{ is } ["the", "inner", "list"] \]
\[ L[2][1] \text{ is } "inner" \]
How do we manipulate them?

Method 1:
Assign a value to a (valid) index

L = [10, 20, 30, 40, 50]
L[2] = 100
L is now [10, 20, 100, 40, 50]
How do we manipulate them?

Method 2:
Built-in methods (call help(list) or visit documentation page)

Example – append()

L = [10, 20, 30, 40, 50]
L.append(60)
L is now [10, 20, 30, 40, 50, 60]
EXAMPLES!